Exhibit 16

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,

Petitioner,

v.

SINGULAR COMPUTING LLC,

Patent Owner.

Patent No. 9,218,156
Filing Date: March 25, 2013
Issue Date: December 22, 2015

Inventor: Joseph Bates
Title: PROCESSING WITH COMPACT ARITHMETIC
PROCESSING ELEMENT

PATENT OWNER'S RESPONSE

Case No. IPR2021-00165

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

Cases

EXHIBIT LIST

Exhibit No.	Description
2001	Defendant Google LLC'S Preliminary Claim Construction Brief in Singular Computing LLC v. Google LLC, No. 1:19-cv-12551 (D. Mass.), dated Jan. 8. 2021.
2002	SEALED
2003	SEALED
2004	SEALED
2005	SEALED
2006	SEALED
2007	SEALED
2008	SEALED
2009	SEALED
2010	SEALED
2011	Dean, J. "The Deep Learning Revolution and Its Implications for Computer Architecture and Chip Design"
2012	Transcription of YouTube Video: Domain Specific Architectures for Deep Neural Networks: Three Generations of Tensor Processing Units (TPUS), In the matter of Google LLC v. Singular Computing LLC, dated June 23, 2021.
2013	SEALED
2014	SEALED
2015	Wang, S, *et al.*, BFLoat16: The Secret to High Performance on Cloud TPUs, dated Aug. 23, 2019.
2016	Jouppi, N, et al., "A Domain-Specific Supercomputer for Training Deep Neural Networks" *Communications of the ACM*, Vol. 63, No. 7, July 2020.
2017	Elias, J. "Google's new TPU 3.0 will be critical to company's success in the post-search world, here is why" *PCMag India Computer and Product Reviews*, May 9, 2018.
2018	"Empowering Businesses with Google Cloud AI" *Cloud TPU*
2019	Patent Owner's Discovery Requests in IPR2021-00165, dated July 6, 2021.
2020	Patent Owner's Discovery Requests in IPR2021-00165, dated June 17, 2021.
2021	Plaintiff's Preliminary Patent-Related Disclosures in Singular Computing LLC v. Google LLC, No. 1:19-cv-12551 (D. Mass.),

Exhibit No.	Description
	dated Sept. 4. 2020.
2022	Infringement Contentions Exs. A-C
2023	Email from E. Hunt to P. Lambrianakos et al., regarding Patent Owner's Discovery Requests, dated June 25, 2021.
2024	Default Protective Order and Standard Acknowledgement for Access to Protective Order Material in IPR2021-00155, -00165, and -00179, dated July 6, 2021.
2025	Transcript of Realtime Conference Call held on July 1, 2021, in IPR2021-00155, -00165, and -00179.
2026	SEALED
2027	SEALED
2028	SEALED
2029	SEALED
2030	SEALED
2031	SEALED
2032	SEALED
2033	SEALED
2034	"Google AI with Jeff Dean," Google Cloud Platform Podcast
2035	SEALED
2036	SEALED
2037	Dean, J. i, "Build and Train Machine Learning Models on our New Google Cloud TPUs" dated May 17, 2017
2038	Stone, Z. "Google's Scalable Supercomputers for Machine Learning, Cloud TPU Pods, are now Publicly Available in Beta" dated May 7, 2019.
2039	SEALED
2040	Metz, C. "Google Rattles the Tech World with a New AI Chip for All" *WIRED,* May 17, 2017.
2041	Wang, S. *et al.*, "BFloat16: The Secret to High Performance on Cloud TPUs" dated Aug. 23, 2019.
2042	SEALED
2043	Deposition of Richard M. Goodin in IPR2021-00155, -00165, and 00179, dated July 30, 2021.
2044	CV of Sunil P. Khatri, Ph.D.
2045	Gorner, M. Google Presentation entitled "TPUs for Developers"
2046	Jouppi, N. *et al.*, "A Domain-Specific TPU Supercomputer for Training Deep Neural Networks" dated Oct. 27, 2020.

Exhibit No.	Description
2047	Google Research People: Jeffrey Dean
2048	XPany Team: Astro Teller
2049	Belletti, F. et al., "Tensor Processing Units for Financial Monte Carlo" dated Jan. 27, 2020.
2050	SEALED
2051	Declaration of Sunil P. Khatri, Ph.D. in IPR2021-00165, dated Aug. 9, 2021.
2052	Declaration of Joseph Bates in IPR2021-00165, dated Aug. 9. 2021.
2053	Declaration of Richard M. Cowell in IPR2021-00165, dated Aug. 9. 2021.

I. INTRODUCTION

The claims of the ’156 Patent are directed to computer systems able to achieve a high level of parallelism and scalability by sacrificing precision. In particular, claim 3 of the ’156 Patent is directed to a computer architecture comprising at least one hundred more low-precision, high dynamic range execution units (LPHDR execution units) than full-precision (32-bit or more) units. By sacrificing precision, individual LPHDR execution units require fewer transistors and occupy less area on a microchip, which in turn allows a computer to have lots of them and specifically in claim 3, many more of them than full precision execution units. This enables such systems to perform more computations per unit of resource (*e.g.*, transistor, area, volume).

At the time of filing, the prevailing view in the prior art was that many applications require high levels of precision and can only be performed by full-precision floating point execution units. Dr. Joseph Bates, however, realized that the conventional wisdom was wrong: he understood that a computer with many LPHDR execution units—and comparatively few full-precision execution units—could perform just as well or better than the full-precision focused systems disclosed in the prior art, even in applications that were thought to require full precision.

Dr. Bates' counterintuitive insight is based on the fact that LPHDR execution units, by virtue of their low transistor count, take up much less space than high-precision ones. Therefore, sacrificing full-precision capability (while maintaining high dynamic range) increases the number of arithmetic units that can fit on a single chip as Dr. Bates explains in the patent specification:

> Despite the common belief among those having ordinary skill in the art that modern applications require high precision processing, in fact a variety of useful algorithms function adequately at much lower precision. . . . [S]uch algorithms may be performed by processors or other devices implemented according to embodiments of the present invention, which come closer to achieving the goal of using a few transistors to multiply . . . thus enabling massively parallel arithmetic computation to be performed with relatively small amounts of physical resources.

Ex 1001, '156 Patent, 7:17-32.

None of the references cited by Google disclose Dr. Bates' fundamental insight about the tradeoff between precision and parallelism. Instead, Petitioner's principal references disclose reduced precision only as a means of reducing power consumption.

In particular, the primary reference cited by Petitioner teaches execution units with full-precision circuitry that can be powered-down when it is not needed and reactivated when it is needed. The full-precision circuitry of these execution units occupies space on the chip, even when it is powered-down.

And more generally, none of the cited art teaches an execution unit that relies on LPHDR operations to make them smaller than conventional full precision execution units, so as to drive the parallelism and scale that characterizes the computing devices of the ’156 Patent which have much larger numbers of LPHDR execution units than full-precision execution units.

Petitioner’s failure to find invalidating prior art is no surprise. Indeed, it is consistent with the overwhelming evidence of objective indicia presented herein—including skepticism, praise, commercial success, nexus, and outright theft of the claimed invention by Petitioner—which collectively confirms the validity of the challenged claims.

More specifically, as explained below, [redacted]

This skepticism was followed by praise from some of the leading minds in the field of computer science. [redacted]

[redacted], Google copied Dr. Bates' patented architecture in building its infringing TPUv2 and TPUv3 products, both of which generated billions of dollars in revenue. The nexus between this commercial success and the claimed invention is demonstrated by Google's own press releases and academic publications in which Google touted the low-precision multiplication units of its TPU products and their reliance on the brain-float 16 (bfloat16) number format to achieve increased parallelism and scale as the chief driver of their success in the marketplace. At the same time, [redacted].

II. THE '156 PATENT

The '156 Patent is entitled "Processing with Compact Arithmetic Processing Element" and issued on December 22, 2015, Ex. 1001. The '156 Patent claims priority, through parent and grandparent applications, to U.S. Provisional Patent Application No. 61/218,691, filed on Jun. 19, 2009.

Dr. Bates recognized that even though then-modern conventional microprocessors contained hundreds of millions of transistors, they could perform only a handful of operations per clock cycle. '156 Patent, 1:55-63. Dr. Bates explained that a large portion of this inefficiency comes from using transistor-intensive full-precision arithmetic units:

> As described above, today's CPU chips make inefficient use of their transistors . . . they deliver great precision, performing exact arithmetic . . . standardized arithmetic with 32 and 64 bit floating point numbers. Many applications need this kind of precision. As a result, conventional CPUs typically are designed to provide such precision, using on the order of a million transistors to implement the arithmetic operations.

Id. at 3:11-3:26. Ex. 2051 ("Khatri"), ¶ 39.

However, Dr. Bates realized that such full-precision, inefficient components were not necessary for all applications, including many valuable ones:

> There are many economically important applications, however, which are not especially sensitive to precision and that would greatly benefit, in the form of application performance per transistor, from the ability to draw upon a far greater fraction of the computing power inherent in those million transistors. Current architectures for general purpose computing fail to deliver this power.

'156 Patent, 3:27-33. Khatri, ¶ 40.

The '156 Patent is thus directed away from prior art computers based on full-precision execution units that take up space and are wasteful of transistors. *Id.*, ¶ 41.

As Dr. Bates further explains in the specification, "[b]ecause LPHDR processing elements are relatively small, a single processor or other device may include a very large number of LPHDR processing elements, adapted to operate in parallel with each other." '156 Patent, 6:56-59. As a result, "embodiments of the present invention may be implemented as any kind of machine which uses LPHDR arithmetic processing elements to provide computing using a small amount of resources (*e.g.*, transistors or volume) compared with traditional architectures." *Id.*, 8:8-12; Khatri ¶ 41.

By using a "very large number" of LPHDR execution units in parallel, computer systems are able to achieve significantly better performance than prior art systems. Because each LPHDR execution unit requires fewer resources (*e.g.*, fewer transistors, less physical volume) than a full-precision execution unit, "there is a large amount of arithmetic computational power per unit of resource. This enables larger problems to be solved with a given amount of resource than does traditional computer designs." '156 Patent, 23:37-44. In particular, the claimed systems "might perform tens of thousand of arithmetic operations per cycle, as opposed to hundreds in a conventional GPU or a handful in a conventional multicore CPU. *Id.*, 23:37-44; Khatri, ¶ 42.

In addition, the ’156 Patent also teaches computer systems in which the number of LPHDR execution units exceeds the number of full precision execution units:

> For certain devices . . . according [to] the present invention, the number of LPHDR arithmetic elements in the device (e.g., computer or processor or other device) exceeds the number, possibly zero, of arithmetic elements in the device which are designed to perform high dynamic range arithmetic of traditional precision (that is, floating point arithmetic with a word length of 32 or more bits).

’156 Patent, 27:52-59; Khatri ¶ 43.

The increased level of compute parallelism and scale in such computer systems is necessarily achieved at the cost of precision—the vast majority of the high dynamic range floating-point operations performed by the device must be performed at low precision. Dr. Bates was the first to understand that sacrificing precision for increased parallelism/scale results in significant performance gains per unit of resource over the prior art. *Id.* ¶ 44.

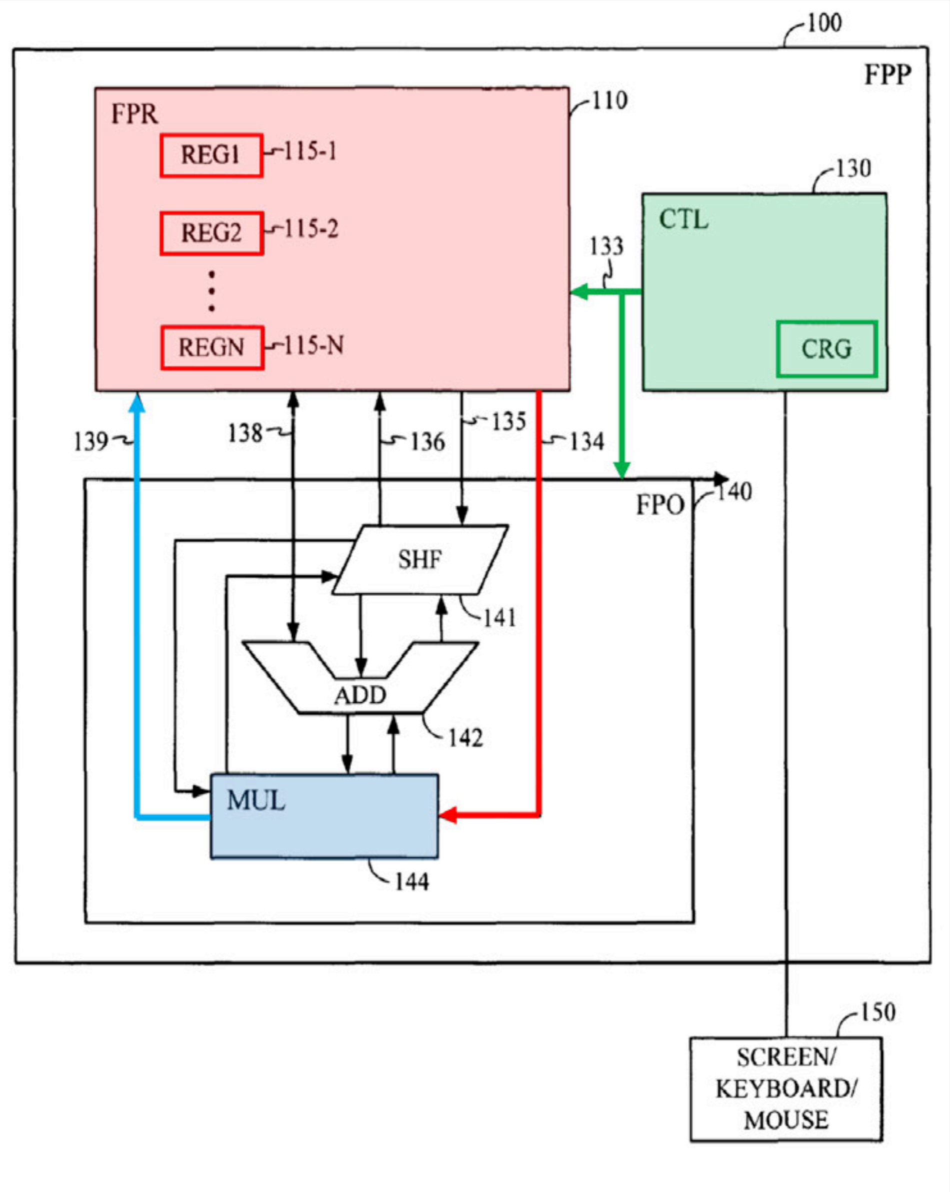
III. OVERVIEW OF THE CITED REFERENCES

A. Dockser

U.S. Pat. Publ. 2007/0203967 (“Dockser”) discloses the use of a full-precision floating point processor (FPP) that can selectably reduce precision in order to reduce its power draw. *See, e.g.*, Ex. 1007, Dockser ¶ [0026]; Khatri, ¶ 45.

Dockser's teaching of an execution unit that is always capable of operating in both full-precision mode (its default) and reduced-precision modes is motivated by the prevalent view in the prior art that while low-precision operation might be acceptable for "certain applications," for a general-purpose processor, full-precision capability is "needed." Dockser ¶¶ [0003, 0018], claims 1, 8, 15, 20; Khatri, ¶ 46.

The components of Dockser's FPP is shown in Figure 1, reproduced below:



Dockser, Fig. 1 (Annotated)

Dockser's FPP has a register file (FPR 110, shaded in **red**) comprising registers (red boxes) that hold IEEE 32-bit full-precision values. It also includes a controller (CTL 130, shaded in **green**) with a control register (**green** box) that can store "subprecision select bits" corresponding to the desired level of precision. Dockser, Fig. 1; *see also id.*, ¶¶ [0017-18]; Khatri, ¶ 47. The FPP performs arithmetic operations (*e.g.*, including addition and multiplication) at full-precision – or at the desired sub-precision using values stored in the register file as operands. *See* Dockser, ¶ [0019]; Khatri, ¶ 47.

If no sub-precision is selected, Dockser's FPP performs arithmetic at "maximum precision" (*i.e.*, 32-bit IEEE full-precision). *See* Dockser, claims 1, 8, 15, 20; *see also id.*, ¶ [0018] ("the floating-point controller 130 *may* be used to select the sub-precision of the floating-point operations . . . ."); Khatri, ¶ 48. Otherwise, the operation is performed at a reduced precision. *Id.* In reduced-precision modes, Dockser's FPP reduces the mantissa size by powering-down some of the cells in the register file that store the mantissa, and also powers down parts of the arithmetic logic circuits (contained, *e.g.*, within the multiplier, shaded in **blue** above) that are not needed for the selected subprecision. Dockser, ¶¶ [0026-27]; Khatri, ¶ 48.

Unlike the '156 Patent, Dockser does not teach systems that include multiple arithmetic units operating in parallel. Ex. 2043, Goodin Dep. 70:13-21; Khatri, ¶ 49. Dockser discloses a single floating-point execution unit in isolation. *Id.* The absence of a parallel teaching is not surprising, given that Dockser's selectable low-precision unit is designed to conserve power in "battery operated devices where power comes at a premium, such as wireless telephones, personal digital assistants (PDA), laptops, game consoles, pagers, and cameras." Dockser, ¶ [0003]; Khatri, ¶ 49.

Because Dockser's unit is capable of full-precision, it would have at least as many transistors and take up at least as much space as a conventional full-precision arithmetic unit, even when operating in a reduced-precision mode. Goodin Dep. 71:6-12; 43:21-45:9. Khatri, ¶¶ 50-51. Indeed, because Dockser's execution unit uses conventional arithmetic units and includes additional control circuits for selecting reduced-precision modes, it is likely *larger* than a conventional full-precision execution unit, making it unsuitable for scaling the compute power of conventional full-precision parallel processing arrays. *Id.*

B. Tong

"Reducing Power by Optimizing the Necessary Precision/Range of Floating-Point Arithmetic" by Jonathan Ying Fai Tong *et al.* Ex. 1008 ("Tong") teaches that using lower-precision arithmetic can reduce power consumption. *See* Tong, 280. Tong is based on the commonly held belief that some applications can only be performed using full-precision arithmetic: "[e]ven though we may be able to assume that most of our operands can be computed successfully in limited precision, it appears inevitable that some fraction of our operands will require full IEEE-standard precision." *Id.*; Khatri, ¶ 52.

Importantly, like Dockser, Tong is focused on reducing power consumption, and does not teach parallel processing systems that include multiple arithmetic units that operate simultaneously, let alone systems having much larger numbers of low-precision units than full-precision units. Khatri, ¶ 53. Each low-precision unit disclosed by Tong is either as large as a full-precision unit or paired with a full precision unit on a 1:1 basis (Tong, 282). Khatri, ¶ 53.

For example, Tong teaches "simply including both full and reduced precision FP units and using appropriate sleep-mode circuit techniques to shut down the unused unit." Tong, 282; Khatri, ¶ 54. In this approach, each low-precision unit is paired with a full-precision unit in a 1:1 ratio, requiring more physical space than a full-precision unit alone. *Id.* Indeed, this approach is presented as an option in situations where silicon real-estate is not at a premium. *See* Tong, 282 ("Given the decreasing cost of silicon area . . . ."); Khatri, ¶ 54.

In another example, Tong also teaches a "digit-serial" multiplication circuit that, using control signals, is operable to perform a reduced-precision operation in a single clock cycle. *Id.*, ¶ 55. The result of this reduced-precision operation, in which an 8-bit operand is multiplied with a 24-bit operand, can be combined with other reduced-precision results over multiple clock cycles in a process called "digit-serial multiplication," yielding a full-precision result. *Id.* When performing low-precision operations, Tong's digit-serial multiplier does consume less power than a traditional full-precision unit. *Id.* However, it occupies *more* physical space than a full-precision multiplier, because it "require[s] extra random logic for control of the multiple passes through the digit serial structure." Tong, 281; *see also* 280 ("The lower precision digit-serial design is slightly larger [than the full-precision execution unit]"), Table V (showing that Tong's digit-serial multiplier occupies ~3% more area than a conventional full-precision multiplier); Khatri, ¶ 55. Therefore, like Dockser's FPP, the execution units taught by Tong are capable of reduced-precision operation, but they are *larger* than a conventional full-precision execution unit, making them unsuitable for increasing the parallelism and compute scale of arrays of full-precision units. *Id.*

As with Dockser, Tong is focused on power-savings, does not even mention increasing compute scale and parallelism, and certainly does not teach a computer comprised of a larger (let alone much larger) number of low-precision units than full-precision units. *Id.*, ¶ 56.

C. MacMillan

U.S. Patent No. 5,689,677 ("MacMillan"), Ex. 1009, is directed to a computer system that includes a host processor and "a plurality of processing elements." Macmillan, 12:39. MacMillan does not describe the capabilities of each "processing element" (PE) in detail, noting only briefly that "[i]nteger and floating point accelerators could be included in each PE." *Id.*, 12:55-56; Khatri, ¶ 57.

Regarding arithmetic precision/imprecision, MacMillan is silent. *Id.*, ¶ 58. It simply teaches that each PE can "perform atomic operations on data values up to *32 bits wide*." MacMillan, 7:8-9[1]; Khatri, ¶ 58. Unlike Dockser and Tong, MacMillan is not focused on minimizing power consumption, and is not specifically tailored for use in portable or mobile devices. *Id.* MacMillan's only reference to power consumption relates to heat dissipation inside the "cabinet" of a workstation. *See* MacMillan, 3:4-6; Khatri, ¶ 58.

IV. THE CHALLENGED CLAIMS AND LEVEL OF SKILL IN THE ART

Petitioner challenges claims 1-8, 16, and 33 (the "Challenged Claims"). Pet. at 1.

A person of ordinary skill in the art (POSA) would be a person with a Bachelor's degree in Computer Science, Electrical Engineering, or Applied Mathematics, with 2 years of academic or industry experience in computer architecture. Khatri, ¶¶ 36-37. This is the same as Petitioner's proposed level of skill, except that Petitioner's level included any number of years of experience more than two. Pet. at 8-9.

[1] All emphasis is added unless otherwise indicated.

V. CLAIM CONSTRUCTION

A. "Low Precision High Dynamic Range (LPHDR) Execution Unit"

Singular's Construction	Google's Construction
"an execution unit that executes arithmetic operations only at low precision and with high dynamic range, wherein 'high dynamic range' and 'low precision' are defined according to the numerical requirements below"	No construction necessary

The Board should construe the term "low precision high dynamic range (LPHDR) execution unit" as "an execution unit that executes arithmetic operations only at low precision and with high dynamic range, wherein 'high dynamic range' and 'low precision' are defined according to the numerical requirements below." This construction captures the plain meaning of the term as read in light of the specification. Khatri, ¶ 59.

In its Institution Decision, the Board stated that "[t]he only limitations on the execution unit recited in the claim are that the execution unit be 'low precision,' 'high dynamic range,' and 'adapted to execute a first operation' meeting certain criteria specified in the imprecision limitation (*i.e.*, a minimum relative error Y for a minimum fraction X of possible valid inputs in a specified dynamic range)." Paper No. 16, 21-22. The Board further accepted, at the institution stage, Petitioner's argument that "the claim does not recite any structural characteristics of the execution unit and does not include any negative limitation precluding the execution unit from performing other types of operations." *Id.* at 22.

As the Board noted however, the execution unit *is* limited to be "low precision" and "high dynamic range." *Id.* at 21. This comports with the plain and ordinary meaning of the term and necessarily excludes full-precision or mixed full and low precision units. Khatri, ¶ 60. To do otherwise would impermissibly read out "low precision high dynamic range" and render the limitation meaningless. *Network-1 Techs., Inc. v. Hewlett-Packard Co.*, 981 F.3d 1015, 1022-23 (Fed. Cir. 2020) (construction of "low level current" was required to give meaning to "low" and noting "the word 'low' in the claim phrase 'low level current' operates to limit the upper boundary of the current level"). The remaining limitations of the claims ("wherein the dynamic range . . . " and "for at least X=5% . . .") properly explain the exact parameters of the low precision and high dynamic range, which do not allow for the execution unit to have full-precision (or high dynamic range) capability. *Biosig Instruments, Inc. v. Nautilus, Inc.*, 783 F.3d 1374, 1378 (Fed.

Cir. 2015) ("When a 'word of degree' is used, the court must determine whether the patent provides 'some standard for measuring that degree.'"); Khatri, ¶ 60.

Indeed, Petitioner's interpretation that any execution unit that "meets each claim's recited error amount" is a "low precision" execution unit would impermissibly read the term "low precision" out of the claims entirely. *Id.*, ¶ 61. That is, under Google's construction or interpretation, the term "at least one first low precision high dynamic range (LPHDR) execution unit adapted to execute" would have exactly the same scope as "at least one execution unit adapted to execute." *Id.* The Board should decline to impermissibly broaden the claims in that fashion. *E.g.*, *Aspex Eyewear, Inc. v. Marchon Eyewear, Inc.*, 672 F.3d 1335, 1348 (Fed. Cir. 2012) (reversing construction of "rearward free end" that read "free" out of the limitation); *Bicon v. Straumann Co.,* 441 F.3d 945, 950-51 (Fed. Cir. 2006) (rejecting construction that rendered claim terms meaningless and noting that "claims are interpreted with an eye toward giving effect to all terms in the claim").

And though Patent Owner agrees with the Board's observation based on the preliminary record that a claim's recitation of an "execution unit adapted to execute a first operation" would normally not preclude that execution unit from performing other types of operations (*see* Paper No. 16 at 22, citing to *KJC Corp. v. Kinetics Concepts, Inc.* 223 F.3d 1351, 1356 (Fed. Cir. 2000)), the rule does not apply when the specification teaches otherwise, as is the situation here. *See TiVo, Inc. v Echostar Commc'ns Corp.*, 516 F.3d 1290, 1303 (Fed. Cir. 2008) ("[T]he question whether 'a' or 'an' is treated as singular or plural depends heavily on the context of its use."); *Harari v. Lee*, 656 F.3d 1331, 1341 (Fed. Cir. 2011) (There is no "hard and fast rule that 'a' always means one or more than one. Instead, we read the limitation in light of the claim and specification to discern its meaning." Here, "[t]he plain language of the claim clearly indicates that only a single bit line is used when accessing a number of cells").

The specification supports this construction in explaining that the "LPHDR execution units" of the '156 Patent are based on a "fundamentally different approach" from the units that incorporate full-precision capabilities as taught in the prior art. '156 Patent, 6:1-5; Khatri, ¶ 62. Unlike the prior art full-precision units, the claimed LPHDR execution units are smaller and have fewer transistors:

> "For example, embodiments of the present invention may be implemented as any kind of machine which uses LPHDR arithmetic processing elements to provide computing *using a small amount of resources* (*e.g.*, transistors or volume) ***compared** with traditional architectures*."

'156 Patent, 8:8-16; Khatri, ¶ 62.

LPHDR execution units utilize less chip real-estate than conventional execution units *precisely because* they do not include full-precision arithmetic circuits:

> "One variety of LPHDR arithmetic represents values from one millionth up to one million with a precision of about 0.1% . . . . One example of an alternative embodiment is to use a logarithmic representation of the values . . . *As a result*, *the area* of the arithmetic circuits *remains* ***relatively*** *small* and a *greater number* of computing elements *can be fit into a given area* of silicon . . . which gives it an advantage for those computations able to be expressed in the *LPHDR* framework."

'156 Patent, 6:8-27; Khatri, ¶ 63.

The small size and low transistor count of each individual LPHDR execution unit is what allows a much larger number of LPHDR execution units to operate in parallel on a single chip:

> Because *LPHDR processing elements are relatively small,* a single processor or other device may include a very large number of LPHDR processing elements, adapted to operate in parallel with each other, and therefore may constitute a massively parallel LPHDR processor or other device.

'156 Patent, 6:56-60; Khatri, ¶ 64. The specification contrasts the small LPHDR units with units, like those inside GPUs, that can operate in both high and low precision. '156 Patent, 5:36-45; Khatri, ¶ 64.

As the specification explicitly states, the fact that "LPHDR execution units" are smaller than full-precision units is not limited to a preferred embodiment, but is an essential aspect of the invention as a whole:

> The discovery that massive amounts of LPHDR arithmetic is useful as a fairly general computing framework, as opposed to the common belief that it is not useful, can be an advantage in any (massively or non-massively) parallel machine design or non-parallel design, not just in SIMD embodiments. It could be used in FPGAs, FPAAs, GPU/SIMT machines, MIMD machines, and *in any kind of machine that uses **compact** arithmetic processing elements* to perform large amounts of computation using a small amount of resources (like transistors or volume).

'156 Patent, 24:8-17; *see also id*, 25:23-28; Title ("Processing with *Compact* Arithmetic Processing Element"); Khatri, ¶ 65.

VI. THE CHALLENGED CLAIMS ARE VALID

A. Ground 1: Claims 1-2 and 16 Are Not Obvious Over Dockser

Dockser does not render obvious any challenged claim of the '156 Patent because each claim of the '156 Patent requires a ***low precision*** high dynamic range LPHDR execution unit. Khatri, ¶ 65. Dockser neither discloses nor renders obvious this limitation. *Id.*

1. Dockser Is Not a Low Precision High Dynamic Range (LPHDR) Execution Unit

As discussed above, the proper construction of "low precision high dynamic range (LPHDR) execution unit" is "an execution unit that executes arithmetic operations only at low precision and with high dynamic range, wherein 'high dynamic range' and 'low precision' are defined according to the numerical requirements below." *Id.*, ¶ 67. Even if the Board chooses to apply the plain meaning of "LPHDR execution unit," a POSA would understand that an LPHDR execution unit is necessarily "low precision" and not capable of high or full precision. *Id.*, ¶ 68.

Dockser, by contrast, is capable of full-precision operation, and is therefore relatively large and "wasteful of transistors" (*See supra* citing '156 Patent 5:23-33); Khatri, ¶ 69. Dockser's 32-bit FPP includes all of the circuitry needed for full precision arithmetic on data in IEEE 32-bit format, and also has additional circuitry and transistors, such as the controller and associated logic, to allow selection of sub-precisions. Dockser, ¶¶ [0019]-[0020]; Khatri, ¶ 69; Goodin Dep. 43:21-45:9. This additional circuitry allows the FPP to perform operations, such as multiplication with selectively reduced precision, while retaining its ability to perform those same operations with full precision. *See* Dockser, ¶ [0026] ("The subprecision select bits <u>*may be*</u> used to reduce the precision of the floating-point operation."); *see also id.* ¶ [0028] (referring to "the full precision mode"); Khatri, ¶

69; Goodin Dep. 37:10-17. Indeed, Dockser explicitly anticipates scenarios in which this subprecision capability will *not* be used, because for certain applications "a greater precision is needed." Dockser, ¶ [0003]; Khatri, ¶ 69. Dockser does not disclose any embodiments of an execution unit without either full-precision or the additional circuitry / transistors for selecting a subprecision, which makes it unsuitable for driving increases in compute scale or parallelism relative to conventional full-precision execution units. *Id.*; Goodin Dep. 38:3-8.

Petitioner argues that Dockser's FPP "is 'low precision' as claimed because 'the precision' of operations is 'reduced' (*e.g.*, [0014]), and because it operates with the minimum imprecision in [1B2]." Pet., 15 (citing Goodin, ¶¶ 208-209). These arguments, which focus only on one particular low-precision mode of the Dockser FPP and ignore the other modes, miss the key point, however, which is that the FPP's full-precision mode makes it too big and too wasteful of transistors to qualify as an LPHDR execution unit. Khatri, ¶ 71.

It is beyond dispute that Dockser has a full precision mode. Khatri, ¶¶ 45-51. Petitioner admitted as much in its Reply to the POPR. Paper No. 14, 2 ("Retaining the prior-art capability for that one 'full' operation does not detract from the purpose of Dockser's design."). Petitioner's expert, Mr. Goodin, agreed that Dockser has a full precision mode. Goodin Dep. 36:16-21. Indeed, Dockser's entire stated purpose is to provide a processor that can perform operations at various precisions, from full precision to lower precision. Dockser, ¶ [0003] ("For general purpose processors, however, the common situation is that for certain applications, *e.g.* generating 3D graphics, a reduced precision may be acceptable, and for other applications, *e.g.* implementing Global Positioning System (GPS) functions, a greater precision may be needed. Accordingly, there is a need in the art for a floating-point processor in which the reduced precision, or subprecision, of the floating-point format is selectable."); Dockser, ¶ [0028] ("The floating-point addition operation in the ***full precision mode*** is performed through a succession of stages . . . ."); Khatri, ¶ 70. Dockser further explains that "the precision for one or more floating-point operations *may* be reduced from that of the specified format." Dockser ¶ [0014]. A POSA, bearing in mind the foregoing excerpts, including Dockser's focus on power-saving and maintaining full precision capability, would thus understand that Dockser is a full-precision 32-bit FPP, capable of operating at lower precisions to save power, that ultimately is too big and too wasteful of transistors to qualify as an LPHDR execution unit. Khatri, ¶ 70; Goodin Dep. at 37:10-38:8.

This POSA understanding would be particularly informed by the '156 Patent's description of a variable precision GPU as background art. Khatri, ¶ 72. Specifically, the specification characterizes a GPU (conventionally, an array of execution units) that supports both half-precision ("16 bit floating point") for "those applications that want it," and full-precision ("32 bit floating point") operations "because they are believed to be needed for traditional graphics applications," as "*GPUs [that] devote substantial resources to 32 . . . bit arithmetic and are wasteful of transistors.*" '156 Patent at 5:36-45; Khatri, ¶ 72. Furthermore, Petitioner does not argue or show that it would have been obvious to modify Dockser's full-precision FPP to become such an LPHDR execution unit. Khatri, ¶ 72.

Accordingly, since the Petition fails to show that Dockser discloses an LPHDR execution unit, and since the Petition sets forth no reason that a POSA would modify Dockser to be an LPHDR execution unit, Petitioner has failed to show that Dockser renders obvious any claims of the '156 Patent. Khatri, ¶ 73.

B. Ground 2: Claims 1-2, 16, and 33 Are Not Obvious Over Dockser in View of Tong

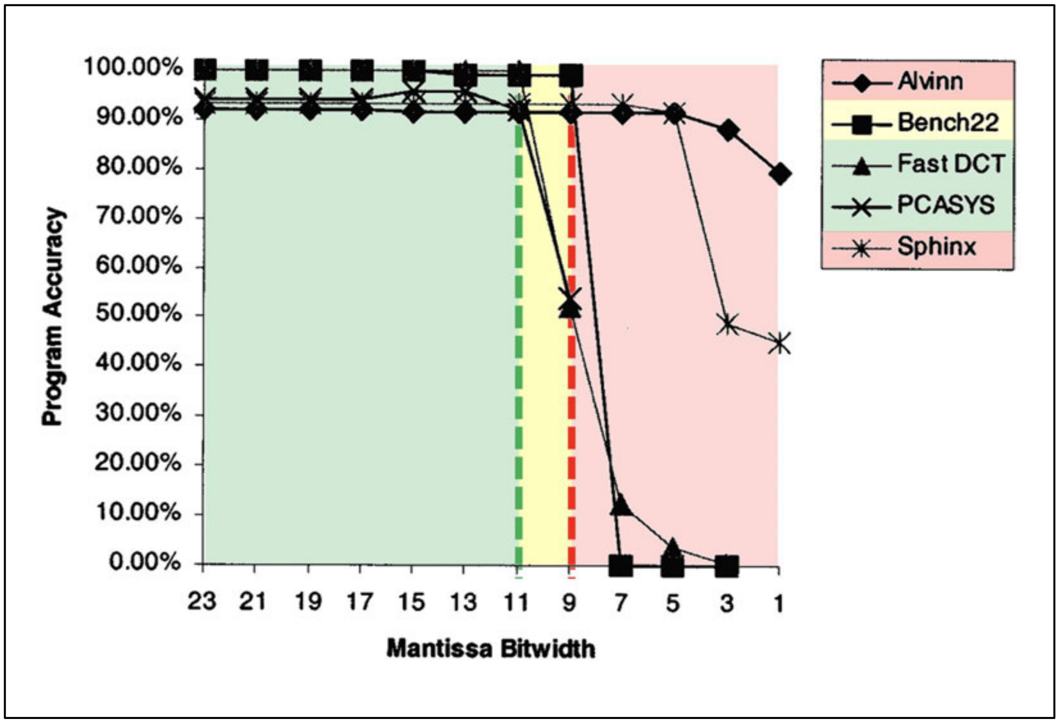
1. Dockser With Tong Does Not Disclose or Render Obvious any Challenged Claim

Petitioner alleges Tong would have motivated a POSA to configure Dockser to operate with 5 mantissa bits of precision. Pet., 45 ("Tong's teachings and empirical tests would have motivated a POSA to configure ***Dockser's FPP*** to operate at the precision levels Tong teaches for particular applications."). Petitioner does not argue that Tong discloses or renders obvious an LPHDR execution unit. *Id.*, 43-47. Since Dockser does not disclose an LPHDR execution unit, and Tong does not remedy that deficiency, the combination of Dockser and Tong (even if it could be or would be used at 5 bits of mantissa) does not disclose or render obvious any Challenged Claim. Khatri, ¶¶ 74-76.

2. Petitioner's Claim that Tong Teaches Using only 5 Mantissa Bits Is Incorrect, which Means a Dockser and Tong Combination Lacks the Imprecision Required to Meet the '156 Patent's X/Y Imprecision Limitations

As an initial matter, Tong teaches that at least 11 mantissa bits are required for consistent performance, even for "programs dealing with human interfaces [that] process sensory data with intrinsically low resolutions." Tong, 278 (noting no "noticeable degradation in accuracy when the mantissa bitwidth is reduced from 23 to 11 bits"); Khatri, ¶ 77. Petitioner makes no effort to show that arithmetic with the "required" 11-bit mantissa is an LPHDR operation and meets the X/Y limitations of the claims.

As for Tong's alleged teaching of using a 5-bit mantissa, Tong's experimental results show that using fewer than 11 mantissa bits unacceptably reduces accuracy for the benchmark applications *that were specifically selected* (from a category of programs) for their high tolerance for imprecision (Khatri, ¶ 78):



Tong, Fig. 6 (Annotated)

These experimental results are shown above, with color annotations denoting the precision levels suitable for general-purpose operation (**green**), those at which a significant percentage of applications begin to produce unacceptable results (**yellow**), and finally, levels of precision that are unsuitable for most testing benchmarks (**red**). Khatri, ¶ 79.

3. There Is No Motivation to Combine Dockser and Tong

Google fails to identify any non-hindsight-based motivation to combine Dockser and Tong. *See* Pet. at 43-46. Google offers no differences between Dockser and the Challenged Claims that would have motivated a POSA to examine any second reference. Moreover, since Dockser and Tong have overlapping functionality, Tong does not disclose any extra capability that would have motivated a POSA to use it with Dockser. Khatri, ¶¶ 81-82.

Petitioner conclusorily states that Tong's teaching—that 5 mantissa bits is sufficient for some applications—would motivate a POSA to configure Dockser to use 5 bits. Pet., 43-44. But there is no evidence to show that a POSA would have chosen Tong's 5-bit examples over the 7, 9, or 11-bit examples. Instead, Petitioner relies on hindsight to choose 5 bits for no purpose other than to meet the claim limitations.

C. Ground 3: Claims 1-8 and 16 Are Not Obvious Over Dockser in View of MacMillan

Petitioner argues that a POSA would combine Dockser with MacMillan by using Dockser's FPP as a "floating point accelerator" within each of MacMillan's PEs. Pet., 52.

Petitioner also uses MacMillan's "Host CPU" to attempt to meet the "at least one first computing device" and CPU limitations of claims 1 and 2, respectively. *Id.*, 52-53. Petitioner further asserts that MacMillan, with Dockser, meets the additional limitation of claim 3: "the number of LPHDR execution units in the device exceeds by at least one hundred the non-negative integer number of execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide." *Id.*, at 51-53.

1. Dockser's FPP is not an "LPHDR execution unit"

As discussed above, Dockser's FPP is not a "low precision high dynamic range (LPHDR) execution unit," as recited in all challenged independent claims under Ground 3, because it does not "execute arithmetic operations "*only* at low precision and with high dynamic range." Khatri, ¶ 83. Petitioner does not argue that MacMillan remedies this deficiency.

2. Dockser and MacMillan Fail to Disclose "wherein the number of LPHDR execution units in the device exceeds by at least one hundred the non-negative integer number of execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide"

Even if the teachings of Dockser and MacMillan were combined to produce a device with, for example, 256 Dockser units operating in parallel, such a device would fail to meet the "exceeds" limitation of claims 3-8. Khatri, ¶ 84.

The "exceeds" limitation requires that the "number of LPHDR execution units in the device exceeds by at least one hundred the non-negative integer number of execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide."[2] *Id.*, ¶ 85. As shown below, Dockser's FPP is adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide. *Id.* Therefore, even if Dockser's FPP were an LPHDR execution unit (which it is not, under the proper proposed construction of that term) the Dockser/MacMillan combination cannot meet claim 3's "exceeds" limitation, because those same "Dockser LPHDR execution units" would also be "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide," meaning that the number of LPHDR execution units will never exceed the number of claim 3 full-precision multiplication execution units. *Id.*, ¶ 86.

(a) Dockser's FPP Is a Claim 3 EU Because It Is Adapted to Execute at Least the Operation of Multiplication on Floating Point Numbers That Are at Least 32 Bits Wide

There is no dispute that Dockser is an execution unit. Pet., 13. And there can be no reasonable dispute that Dockser is adapted to execute at least the

[2] For clarity, Patent Owner refers to "execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" as "claim 3 full-precision multiplication execution units."

operation of multiplication on floating point numbers that are at least 32 bits wide under the plain meaning of the term.[3] Goodin Dep. 23:17-22; Khatri ¶¶ 90-91.

Dockser discloses that its FPP takes input operands that are 32 bits wide and stores them in IEEE-754 format. *See e.g.*, Dockser, ¶ [0017] ("Each register location 200 is configured to store a 32-bit binary floating-point number, in an IEEE-754 32-bit single format.") Khatri, ¶ 92. Dockser also discloses that its FPP includes "a floating-point multiplier (MUL) 144 configured to execute floating-point multiply instructions." Dockser, ¶ [0019]; Khatri, ¶ 92. These points are not in dispute; Petitioner and its expert both admitted as much. Reply at 2; Goodin Dep. at 36:16-21; 38:18-39:15.

Nothing more is required for a Dockser FPP to be "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide." *See In re Man Machine Interface Techs. LLC*, 822 F.3d 1282, 1286 (Fed. Cir. 2016) ("[T]he phrase 'adapted to' generally means 'made to,' 'designed

[3] Petitioner and Patent Owner agree that "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" means that the execution unit performs multiplication in "traditional"—*i.e.*, full—precision. Pet., 52.

to,' or 'configured to,' though it can also be used more broadly to mean 'capable of' or 'suitable for.'"); Khatri, ¶ 92.

Therefore, even if Dockser's FPP is an "LPHDR execution unit" (as Petitioner incorrectly argues), the Dockser/MacMillan device would still fail to satisfy the "exceeds" limitation. Khatri, ¶ 93. In one example, the Dockser/MacMillan device would have 256 LPHDR execution units (counting each Dockser FPP as an "LPHDR execution unit"). However, because each Dockser FPP is also "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide," the Dockser/MacMillan device would therefore also have at least 256 execution units "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide." *Id.* As a result, the Dockser/MacMillan would not meet the "exceeds limitation." *Id.*

(b) Petitioner's Interpretation of "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" Is Inconsistent with the Specification and Wrong

Petitioner attempts to put Dockser's FPP outside of the "exceeds" limitation by arguing that units are claim 3 full-precision multiplication execution units ***only if*** they are "'traditional precision' execution units that do not 'sometimes' produce results different from the correct traditional-precision result." Pet., 52. The Board preliminarily accepted this argument.

Petitioner's implicit construction of "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" is inconsistent with the plain and ordinary meaning of the claim and at odds with the specification. Khatri, ¶¶ 94-95. The Board should thus reject it.

A POSA would understand this term to include all units in the device that are designed to perform "multiplication on floating point numbers that are at least 32 bits wide," even if those units can perform other operations as well.[4] *Id.*, ¶ 95.

[4] While the term at issue here includes the phrase "adapted to," Patent Owner's construction does not depend on any particular construction of this term. Depending on context, "adapted to" can be construed broadly (to mean "capable of") or narrowly (to mean "designed to"). *See In re Giannelli,* 739 F.3d 1375, 1379 (Fed. Cir. 2014); *see also In re Man Machine Interface Techs. LLC*, 822 F.3d at 1286 (citations omitted). In this situation, both of these constructions boil down to the same thing: if a unit is "capable of" performing 32-bit multiplication, then it is "designed to" perform 32-bit multiplication—the ability to perform full-precision multiplication is not something that happens by accident.

First, Petitioner's interpretation directly contradicts the claim language itself which encompasses every execution unit adapted to perform "*at least*" the operation of 32-bit multiplication. Khatri, ¶ 96. Petitioner not only ignores this inclusive language, but also proposes a construction that would explicitly *exclude* units adapted to perform additional operations. *Id.*

Second, Petitioner's interpretation contradicts the teachings of the specification. *Id.*, ¶ 97. For example, while the specification does not disclose an execution unit that can be both a claim 3 full-precision multiplication execution unit and an LPHDR execution unit, it *does* disclose an execution unit that can be both a claim 3 full-precision multiplication execution unit and *a* unit (which is not an LPHDR execution unit, as explained below) that 'sometimes' produces results different from the correct traditional-precision result:

> "When a graphics processor includes support for 16 bit floating point, that support is alongside support for 32 bit floating point . . . That is, the 16 bit floating point format is supported for those applications that want it, but the higher precision formats also are supported because they are believed to be needed for traditional graphics applications and also for so called "general purpose" GPU applications. Thus, *existing GPUs devote substantial resources to 32 . . . bit arithmetic and are wasteful of transistors* . . . ."

'156 Patent, 5:36-45; Khatri, ¶ 97.

A POSA would understand that the "graphics processor" described in the passage above includes multiple execution units operating in parallel, each of which can be configured to perform both full-precision ("32 bit floating point") and half-precision ("16 bit floating point") operations. Khatri, ¶ 98. Such execution units are *both* "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" *and* "capable of performing *"*operations that 'sometimes' produce results different from the correct traditional-precision result" (*i.e.*, half-precision operations that produce results in the "16 bit floating point" format often referred to as "fp16," which is not to be confused with the earlier-mentioned bfloat16/bf16 (*brain* float 16) format used by Google's products). *Id.*

As the passage explains, these execution units are "wasteful of transistors" because, like the execution units of Dockser and Tong, they retain the *capability* of performing 32-bit multiplication, even if they can be configured to "sometimes" operate at lower precisions. *Id.*, ¶ 99. A POSA would understand that such conventional execution units having full-precision *capability* (whether or not they execute other precision operations) are precisely what the "adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" limitation of claim 3 is intended to capture, and more importantly, precisely what distinguishes these conventional execution units from "LPHDR execution units" that perform *only* low-precision arithmetic operations (and thus use "a small amount of resources") in claim 3. '156 Patent, 8:9-12; Khatri, ¶ 99.

More generally, the specification describes claim 3 full-precision multiplication execution units only by reference to their "high dynamic range arithmetic of traditional precision." *See* '156 Patent at 28:9-16; *see also id.* at 28:9-67; Khatri, ¶ 100. Nowhere does the specification ever preclude a claim 3 full-precision multiplication execution unit from being able to produce incorrect results. *Id.* Petitioner imports that requirement from a completely different section of the patent, taken completely out of context. *See* Pet., 54 (citing '156 Patent, 26:61-27:4); Khatri, ¶ 100. That section uses "sometimes" to describes the imprecision limitation of an LPHDR execution unit, as expressed in the X=5% limitation of the Challenged Claims. That section says nothing about claim 3 full-precision multiplication execution units. *See* '156 Patent, 26:61-27:4; Khatri, ¶ 100.

Petitioner's argument tacitly hinges on the idea that if a component is an LPHDR execution unit (*i.e.*, the Dockser FPP as construed by Petitioner, not as construed by Patent Owner), it cannot also be a claim 3 full-precision multiplication execution unit. Pet., 54. That is, Petitioner tacitly argues that one device—the Dockser FPP—cannot meet two limitations in a claim. But the law is clear that the same structure can meet two limitations. *See, e.g., Applied Med. Res.*

Corp. v. U.S. Surgical Corp., 448 F.3d 1324, 1333 n.3 (Fed. Cir. 2006) ("[T]he use of two terms in a claim requires that they connote different *meanings,* not that they necessarily refer to two different *structures.*") (emphasis in original). Therefore, even if the Dockser FPP were an "LPHDR execution unit" (which it is not), it would *still* be a claim 3 full-precision multiplication execution unit for the reasons explained above.

(c) A POSA Would Not Combine Dockser with MacMillan

Even if the combination of Dockser and MacMillan had all the elements of the Challenged Claims – which it does not, for the reasons explained above – a POSA would not make that combination because it would not be operable for its intended purpose. Khatri, ¶ 101.

Dockser is focused on the objective of reducing power consumption and is not in any way concerned with the objective of achieving high compute scale, while MacMillan is focused on a parallel architecture that increases computational power, and not focused on reducing power consumption (indeed, MacMillan's only reference to power consumption relates to heat dissipation inside the "cabinet" of a workstation). *See* MacMillan, 3:4-6; Khatri, ¶ 102.

Incorporating Dockser's FPPs into MacMillan would also defeat MacMillan's stated objective of achieving a high-scale SIMD computer architecture at "lower system cost." *See* MacMillan, 5:58-59 ("The invention of this shared memory results in lower system cost . . . ."); Khatri, ¶ 103. As explained above, Dockser's FPPs are even larger than traditional full-precision execution units because of the control circuitry needed to implement the selectable subprecision modes. *Id.*; Goodin Dep. 70:6-12 (explaining that Dockser does not reduce space compared to conventional units), 43:21-45:9 (additional circuitry required to reduce precision). As a result, replacing the full-precision execution units of MacMillan with Dockser FPP units would require additional circuitry and chip space and would therefore *increase* costs, while providing no benefit. Khatri, ¶ 104. A POSA would not be motivated to destroy the objectives of the prior art. *Trivascular, Inc. v. Samuels*, 812 F.3d 1056, 1068 (Fed. Cir. 2016); *Chemours Co. FC, LLC v. Daikin Indus., Ltd.*, No. 2020-1289, 2021 WL 3085514, at *4-*5 (Fed. Cir. July 22, 2021).

3. Dockser/MacMillan Fails to Disclose or Render Obvious "at least one first computing device adapted to control the operation of the at least one first LPHDR execution unit"

Claim 1 (and therefore its dependent claims 2-8) require "at least one first computing device adapted to control the operation of the at least one first LPHDR execution unit." Claim 2 further requires that the computing device "comprises at least one of a central processing unit (CPU), a graphics processing unit (GPU), a field programmable gate array (FPGA), a microcode-based processor, a hardware sequencer, and a state machine." Petitioner identifies MacMillan's "Host CPU" as the "computing device adapted to control the operation of the at least one first LPHDR execution unit." Pet., 49-51; Khatri, ¶ 87.

However, MacMillan makes clear that the SIMD Controller, not the Host CPU, controls the operation of the PEs. MacMillan, 13:35-36 ("Since the PEs operate under the control of the SIMD Controller . . . ."); 8:1-10 (explaining that SIMD Controller "contains the program counter used to step through the list of instructions that may include instructions to be executed by the PEs"); 10:65-11:5 ("The SIMD Controller 252 then executes the SIMD program."); 10:3-16; Khatri, ¶ 88. Indeed, while the PEs are executing, the SIMD Controller takes control of the host bus; thus there is no way for the Host CPU to control (or even communicate with) the PEs. MacMillan, 10:53-11:5; Goodin Dep. 126:3-129:6; 142:14-143:21; Khatri, ¶ 88. Not only is the Host CPU not disclosed as controlling the PEs, it is merely optional. Goodin Dep., 116:6-18; Khatri, ¶ 88.

Petitioner's expert, Mr. Goodin, agreed that the SIMD Controller controls the operation of the PEs.

> Q. And MacMillan says that the processor elements are controlled by the SIMD controller?
>
> A. Yes, that's correct, that the processing elements are controlled by the SIMD controller 252.

Goodin Dep., 124:18-22.

Since Petitioner has provided no other argument with respect to the "computing device adapted to control" limitation as recited in claims 1-8, Petitioner has failed to show that claims 1-8 are obvious over Dockser and MacMillan. *See* Pet., 52-53, 56-57. Khatri, ¶ 89.

D. Ground 4: Claims 1-8, 16, and 33 Are Not Obvious Over Dockser in View of Tong and MacMillan

As discussed above with respect to Grounds 2 and 3, the combination of Dockser, Tong, and MacMillan does not disclose or render obvious an LPHDR execution unit. Similarly, as set forth with respect to Ground 3, the combination does not disclose or render obvious the "at least one first computing device" limitation. Finally, as set forth with respect to Ground 2, Tong does not teach using only 5 bits of mantissa. Petitioner sets forth no further argument or evidence with respect to any of these limitations.

1. The Dockser/MacMillan System as Modified by Tong Would Not Meet the "Exceeds" Limitation

Even under Petitioner's construction of "LPHDR execution unit," the combination of Dockser, MacMillan, and Tong would still fail to satisfy the "exceeds" limitation. As explained above, each Dockser FPP, even when operating at 5-bit precision, is an "execution unit[] adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide." Thus, even under Petitioner's construction, a device with 256 (or any number) Dockser FPPs operating at 5-bit precision would have *at least* as many full-precision units as "LPHDR execution units." Khatri, ¶¶ 105-06.

2. A POSA Would Not Combine Tong, Dockser, and MacMillan

A POSA would not be motivated to combine the teachings of Dockser, MacMillan, and Tong. As explained above, a person of ordinary skill in the art would not have been motivated to incorporate Dockser's FPP into MacMillan, nor to combine Dockser with Tong. Khatri, ¶ 107. Tong supplies no additional motivation to combine the teachings of Dockser and MacMillan.

VII. PETITIONER'S "ALTERNATIVE" ARGUMENT REGARDING CLAIMS 3-8 FAILS

Recognizing that a system with Dockser's FPPs can never meet the "exceeds" limitation, Petitioner posits that it would be obvious to completely redesign Dockser's FPP to carve out all support for its basic and default mode, namely full-precision operation (*see supra* Section VI.A.1). Specifically, Petitioner posits that it would be obvious to cleave off parts of Dockser's 32-bit register file to achieve some unspecified lesser bit-width, and further perform some unspecified modification to Dockser's multiplier logic "to have only as many logic elements as needed to multiply mantissas of the reduced bitwidth."[5] Pet., 58. Petitioner further notes that Tong suggests a precision of "between 5 and 11 mantissa fraction bits" for "five signal processing applications." Pet., 57-58. And because two of the many applications of an embedded system disclosed in MacMillan are "signal processing" and "voice recognition," Petitioner states that a POSA would use "Dockser's FPPs in MacMillan's architecture with Tong's precision levels." *Id.*

Petitioner is wrong. Far from being obvious, a POSA would not have been motivated to re-engineer the FPP of Dockser to remove its 32-bit capability based on a single, cherrypicked data-point in Tong, and then incorporate the re-engineered Dockser unit into the system of MacMillan because both MacMillan

[5] In its decision instituting trial, the Board suggested that Dockser would not meet the adapted-to limitation because "it has registers and multipliers with less than 32 bits." Paper Nos. 16, 37. However, as Petitioner recognizes (Pet., 59-60), and its expert admits (Goodin Dep., 26:16-23, 38:18-39:15), Dockser explicitly states that its registers and multipliers are 32-bits wide. *E.g.*, Dockser, [¶0017] ("Each register location 200 is configured to store a 32-bit binary floating-point number . . . ."); Dockser, ¶ [0019] (describing multiplication on the 32-bit numbers stored in the registers).

and Tong happen to mention "signal processing." Khatri, ¶¶ 108-09. Petitioner concocted this convoluted and implausible scenario in an attempt to sidestep the Dockser FPP's full-precision capabilities.

A. A POSA Would Not Be Motivated to Combine Dockser, MacMillan, and Tong under Petitioner's "Alternative" Argument

Petitioner's assertion that a POSA would have been motivated by Tong to remove the full-precision capabilities of Dockser is wholly without merit.

Petitioner's argument is based on a fragment of a sentence in Tong, taken out of context to insinuate that full-precision operation "is not essential" to the functioning of a Tong or Dockser system. Pet., 58.

In fact, Tong teaches the opposite. Khatri, ¶¶ 110-111. Tong admits that there are "scientific programs" that "require a huge amount of precision" (Tong, 279), and broadly teaches that "it appears *inevitable* that some fraction of our operands will require full IEEE-standard precision." Tong, 280; Khatri, ¶ 111. Instead, like Dockser, Tong teaches systems that always have *both* full- and reduced-precision capabilities. *See*, *e.g.*, Tong, 282 (even when describing a device that has reduced precision units, it describes that system as "including *both* full and reduced precision FP units and using appropriate sleep-mode circuit techniques to shut down the unused unit."); Khatri, ¶ 111. With this teaching in hand, a POSA would not cleave off parts of Dockser's 32-bit register file to achieve some unspecified lesser bit-width smaller execution unit, without also importing the Tong "full precision FP." *See generally* Tong; Khatri, ¶ 111.

In view of the above, it is clear that a person of ordinary skill would not be motivated by Tong to remove the full-precision capabilities of the Dockser units. Khatri, ¶ 112. On the contrary, Tong would only have *reinforced* the teaching of Dockser that, while reduced-precision might be a viable option in certain circumstances, an execution unit should retain the ability to operate at full-precision because many applications require full precision. *See*, *e.g.*, Dockser, ¶ [0003]; Khatri, ¶ 112. Similarly, Petitioner's "alternative" argument runs contrary to the teachings of Dockser and MacMillan.

Specifically, Dockser's objectives are fundamentally directed away from Petitioner's proposed "alternative" combination. As discussed above, it is undisputed that Dockser is devoted solely to a general processor with selectable precision. Khatri, ¶ 113.

Nowhere, however, does Dockser teach, or even suggest, removing its full-precision capabilities. Khatri, ¶ 114; Goodin Dep. 38:3-8. A Dockser FPP is described as always needing to support a range of selectable precisions including full precision. Adjusting Dockser by removing its full-precision capacity violates a central tenet of Dockser—to always be able to execute full-precision operations. Khatri, ¶ 114; Dockser, ¶ [0003]. Indeed, Mr. Goodin explained that "Dockser

requires selectable subprecision." Goodin Dep. at 56:2-12; 57:4-12. *See also id.* at 38:3-8 ("I don't believe Dockser discloses any embodiments without floating-point – without selectable precision."). Mr. Goodin also explained that he did not provide any opinions where Dockser was limited to a single subprecision. *Id.* 69:17-70:18 ("That certainly isn't part of my opinions in my declaration."). It is not obvious to remove a reference's inventive concept—in this case supporting a range of selectable precisions including full precision—in an attempt to meet a challenged claim. *Chemours Co.*, 2021 WL 3085514, at *5 (reversing Board judgment of obviousness where proposed combination "would necessarily involve altering the inventive concept" of the prior art reference). Cleaving off circuitry from Dockser in an alleged combination with Tong contradicts the inventive concept of Dockser (as well as of Tong, as described in the previous subsection) and could only be the product of impermissible hindsight. Khatri, ¶ 114.

As for MacMillan, it warns that "[t]o meet the cost objectives, the SIMD capabilities ***should not add significant complexity*** to the architecture of a computer system for personal use." MacMillan, 5:42-44. Petitioner's proposed combination, which requires special, customized registers, logic elements, arithmetic units, and programming models (see next subsection), would increase manufacturing costs and goes directly against the teachings of MacMillan which relies on operating with conventional components to reduce cost. See Khatri, ¶ 115. *E.g.*, MacMillan*,* 6:24-26, 34-36.

B. Google Has Failed to Show That the "Alternative" Combination Would Meet the Imprecision Requirements of Claims 3-8

Petitioner has not even attempted to show that the resulting combination would meet the imprecision limitations. Petitioner merely states that the "selected precision levels are unchanged from Grounds 1-3." Pet., 59. However, as shown above, Tong suggests 11 bits of precision for specific signal processing applications, and Petitioner provides no analysis of whether 11 bits of precision would meet the imprecision limitation. *See id.* Khatri, ¶ 116.

C. POSA Would Not Have Recognized the Utility of Petitioner's "Alternative" Combination of Dockser, MacMillan, and Tong

Finally, a POSA would not have recognized the utility of Petitioner's "alternative" combination of Dockser, MacMillan, and Tong because the combined prior art references do not teach or suggest that it would be possible to write programs that run efficiently on a Dockser/MacMillan/Tong device. Khatri, ¶ 117.

As the '156 Patent explains, "programmers have come to think in terms of high precision and to develop algorithms based on the assumption that computer processors provide such precision . . . ." '156 Patent, 5:63-67. The notion that low-precision computers can be programmed "is not obvious, and in fact has been viewed as clearly false by those having ordinary skill in the art." *Id.*, 7:31-35. A POSA at the time of the invention, with a Bachelor's degree and only two years of experience, would not have known how to efficiently write a program in low precision. Khatri, ¶¶ 118-19. Indeed, MacMillan specifically warns to avoid the "need for reprogramming." MacMillan, 2:15-16; Khatri, ¶ 120.

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

However, the ’156 Patent explains that “in fact a variety of useful and important algorithms can be made to function adequately at much lower than 32 bit precision in a massively parallel computing framework, and certain embodiments of the present invention support such algorithms, thereby offering much more efficient use of transistors, and thereby provide improved speed, power, and/or cost, compared to conventional computers.” ’156 Patent, 7:37-44. For example, the ’156 Patent teaches that the “Kahan method” can be used to reduce the accumulation of errors when summing multiple low-precision results. *Id.*, 21:65-22:14. The ’156 Patent also explains how the additional computational power of the claimed invention can be efficiently used by processing data in a “pipelined” fashion. *See id.*, 20:57 to 21:15. The specification also describes in detail several example programs that use unconventional techniques to perform various computational tasks correctly and efficiently on the claimed low-precision systems. *See generally id.*, 17:30-23:34. Khatri, ¶ 120.

Without these teachings of the ’156 Patent, a POSA would not have been motivated to combine MacMillan and Dockser with Tong according to Petitioner’s “alternative” argument because it would not have been clear to a POSA that it would be possible to program such a device to execute operations efficiently and without accumulating errors. *Id.*, ¶¶ 121-22.

VIII. OBJECTIVE INDICIA OF NON-OBVIOUSNESS SUPPORT THE VALIDITY OF THE CLAIMS

Overwhelming evidence of objective indicia of non-obviousness—including skepticism, praise, unexpected results, commercial success, and outright copying of the claimed invention—confirms that the challenged claims are not obvious over the prior art references cited by Petitioner.

A. Legal Standard

"The objective indicia of non-obviousness play an important role as a guard against the statutorily proscribed hindsight reasoning in the obviousness analysis." *WBIP, LLC v. Kohler Co.*, 829 F.3d 1317, 1328 (Fed. Cir. 2016). "[W]e have repeatedly stressed that objective considerations of non-obviousness must be considered in *every* case." *Id.* Indeed, "[s]econdary considerations evidence can establish that 'an invention appearing to have been obvious in light of the prior art was not' and may be 'the most probative and cogent evidence in the record.'" *Apple Inc. v. Int'l Trade Comm'n*, 725 F.3d 1356, 1366 (Fed. Cir. 2013).

Objective indicia of non-obviousness include skepticism, praise, unexpected results, copying, and commercial success. *See Kinetic Concepts, Inc. v. Smith & Nephew, Inc.*, 688 F.3d 1342, 1367 (Fed. Cir. 2012); *see also Mintz v. Dietz & Watson, Inc.*, 679 F.3d 1372, 1379 (Fed. Cir. 2012).

"For secondary considerations to have probative value, the decision maker must determine whether there is a nexus between the merits of the claimed invention and the secondary considerations." *Ashland Oil, Inc. v. Delta Resins & Refractories, Inc.*, 776 F.2d 281, 306 n.42 (Fed. Cir. 1985).

B. Google's Initial Skepticism of the Invention Shows Non-Obviousness

"Evidence of industry skepticism weighs in favor of non-obviousness. If industry participants or skilled artisans are skeptical about whether or how a problem could be solved or the workability of the claimed solution, it favors non-obviousness. Doubt or disbelief by skilled artisans regarding the likely success of a combination or solution weighs against the notion that one would combine elements in references to achieve the claimed invention." *WBIP*, 829 F.3d at 1335-36; *see also e.g.*, *Neptune Generics, LLC v. Eli Lilly & Co.*, 921 F.3d 1372, 1377-78 (Fed. Cir. 2019) ("Evidence of industry skepticism is a question of fact that weighs in favor of non-obviousness.").

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted] is evidence that the Challenged Claims are not obvious. There is a clear nexus between these statements and the Challenged Claims. Khatri, ¶ 123. They were written in response to Dr. Bates's disclosure of the invention, as explained above. *Id.* Further, they express skepticism about utility of the low-precision arithmetic performed by the large numbers of low-precision execution units, which is a key feature of the '156 Patent claims.

C. Praise From Leaders in the Field Shows Non-Obviousness

Praise of the invention by others is another objective indicator of non-obviousness. *Mintz*, 679 F.3d at 1379.

As Dr. Bates' ideas circulated more broadly [redacted]

[redacted]

There is a clear nexus between the praise cited above and the Challenged Claims. It relates directly to [redacted]

D. The Invention Is Not Obvious Because It Yields Unexpected Results

Unexpected results are further evidence of non-obviousness. *See Ortho-McNeil Pharm., Inc. v. Mylan Labs., Inc.*, 520 F.3d 1358, 1365 (Fed. Cir. 2008) ("Of particular importance beyond the prima facie analysis, this court also detects evidence of objective criteria showing nonobviousness. Specifically, the record shows powerful unexpected results . . . ."); *see also Neptune*, 921 F.3d at 1377-78; *Kao Corp. v. Unilever U.S., Inc.*, 441 F.3d 963, 970 (Fed. Cir. 2006).

[redacted]

[redacted]

[redacted],

[redacted]:

[redacted]

The fact that these results were unexpected is further confirmed by [redacted]

Similarly, Dr. David Patterson of Google candidly remarked that he was "genuinely surprised" that the bfloat16 format provided useful results and improved performance, scaling, and power use. Ex. 2012. Dr. Patterson is a "skilled artisan" and acclaimed expert in the field of computer science and computer architecture. Dr. Patterson's comments have a nexus to the '156 Patent because, as shown below, the TPUv2 product is coextensive with the claims. Moreover, his comments further have a nexus because they are directed exactly to the key features of the '156 Patent claims: low precision high dynamic range arithmetic.

The fact that the '156 Patent's invention produces unexpected results is further supported by Dockser and Tong. For example, as explained above, Dockser teaches that for "general purpose processors, however, the common situation is that . . . a greater precision may be needed." Dockser, [0003]; Khatri, ¶¶ 124-25. Similarly, Tong notes that in order to be "generally useful," an execution unit must be capable of full-precision arithmetic. Tong, 280; Khatri, ¶¶ 124-25.

The fact that the prior art references cited by Petitioner in this proceeding teach that an execution unit without full-precision capability is not "generally useful," and Google's initial reaction to the '156 Patent's invention, shows that the high performance of the '156 Patent's invention across a wide variety of applications was unexpected. *Id.* This unexpected result weighs toward non-obviousness.

E. Google's Copying of the Invention Shows Non-Obviousness

"The fact that a competitor copied technology suggests that it would not have been obvious." *WBIP*, 829 F.3d at 1336.

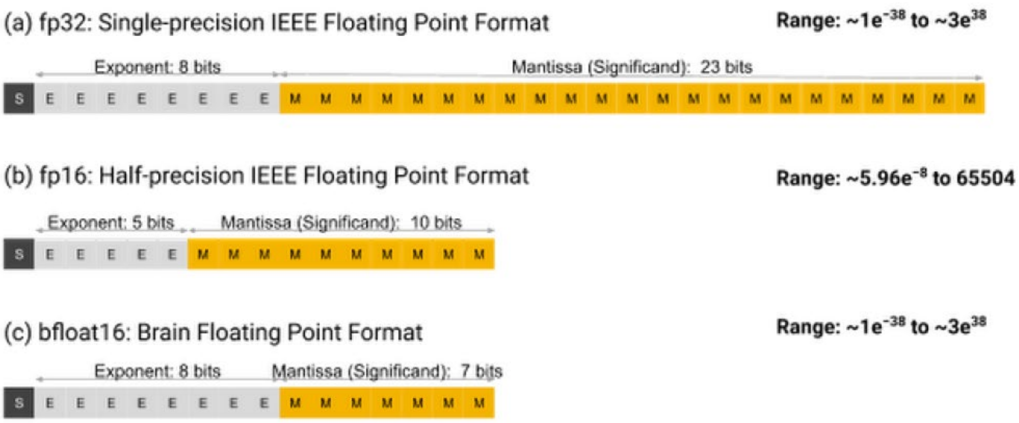
As Dr. Bates was interacting with Google in 2013-14, Google had identified a "daunting and scary" problem: in order to use artificial intelligence for speech recognition, it would need to "double the computing footprint of Google just to support, like, a slightly better speech recognition model for a modest fraction of [its] users." Ex. 2034, 6.

Despite [redacted] Google solved this problem by copying Dr. Bates' invention. Specifically, Google adopted the invention in their TPUv2 and TPUv3 products, which are coextensive with the Challenged Claims (*see* Section VIII, *infra*). Indeed, Dr. Bates's invention is central to the operation of the TPUv2 and TPUv3 products, which use execution units designed to perform operations using a low-precision "floating point format called bfloat16" in order to achieve increased parallelism and scale:

> Because multipliers for the bfloat16 format require so much less circuitry, it is possible to put more multipliers in the same chip area and power budget, thereby meaning that ML accelerators employing this format can have higher flops/sec and flops/Watt, all other things being equal.

Ex. 2011 at 9; Khatri, ¶¶ 126-27.

Google describes the computing enabled by the bfloat16 format as "[t]he secret to high performance on Cloud TPUs," noting that "Cloud TPU v2 and Cloud TPU v3 primarily use bfloat16 in the matrix multiplication unit (MXU)" and that "inside the MXU, multiplications are performed in bfloat16 format." Ex. 2041 at 2. The bfloat16 format maintains high dynamic range by keeping the same 8 bit exponent as found in 32-bit IEEE floating point. *Id*. However, it reduces precision significantly by using only a 7-bit mantissa, as opposed to the 23-bit mantissa of IEEE 32-bit floating point, which is why it requires less circuitry. *Id.*



(a) fp32: Single-precision IEEE Floating Point Format Range: $\sim1e^{-38}$ to $\sim3e^{38}$

Exponent: 8 bits
Mantissa (Significand): 23 bits
S E E E E E E E E M M M M M M M M M M M M M M M M M M M M M M M

(b) fp16: Half-precision IEEE Floating Point Format Range: $\sim5.96e^{-8}$ to 65504

Exponent: 5 bits
Mantissa (Significand): 10 bits
S E E E E E M M M M M M M M M M

(c) bfloat16: Brain Floating Point Format Range: $\sim1e^{-38}$ to $\sim3e^{38}$

Exponent: 8 bits
Mantissa (Significand): 7 bits
S E E E E E E E E M M M M M M M

Id. at 1. A nexus with the claimed invention is presumed for secondary considerations relating to Google's TPUv2 and TPUv3 products, because these products are coextensive with the Challenged Claims, as demonstrated in detail in Section VIII, *infra*. *See WBIP*, 829 F.3d at 1329 ("[T]here is a presumption of nexus for objective considerations when the patentee shows that the asserted objective evidence is tied to a specific product and that product 'is the invention disclosed and claimed in the patent.'").

A nexus between Google's products and the claimed invention is further demonstrated [redacted]

[redacted]

[redacted]

Google's copying therefore provides clear evidence that the Challenged Claims are not obvious. *E.g.*, *Liqwd, Inc. v. L'Oreal USA, Inc.*, 941 F.3d 1133, 1139 (Fed. Cir. 2019) (disclosure of patented method and later adoption of technology supported finding of copying); *see also Power Integrations, Inc. v. Fairchild Semiconductor Int'l, Inc.* 711 F.3d 1348, 1369 (Fed. Cir. 2013).

F. The Commercial Success of Google's TPUv2 and TPUv3 Shows Non-Obviousness

Commercial success is an objective indicator of non-obviousness. *See, e.g., id.* at 1368. The commercial success of Google's TPUv2 and TPUv3 products, which are coextensive with the Challenged Claims, supports a finding of non-obviousness.

Since the release of TPuv2 in 2017, and TPUv3 in 2018, Google has leveraged the abundance of processing power made available by Dr. Bates' invention—its "secret to high performance" over its competitors—to power all of its major products and services, including Search, Translate, Photos, Assistant, and Gmail. Ex. 2018. *See also* Exs. 2040, 2015.

Using hundreds of thousands of these low-precision TPU v2 and v3 devices to provide these AI services in its data centers throughout the U.S., Google has saved at least $10 billion dollars. Ex. 2017. Thus, Google's use of Dr. Bates' low-precision computer throughout its data centers has resulted in enormous commercial success for Google.

This commercial success is directly tied to the claimed invention. Google's TPUv2 and TPUv3 products are coextensive with the Challenged Claims, as demonstrated in detail in Section VIII, *infra*. Further, the success of these products in the marketplace is due in large part to their use of the claimed invention, which allows them to achieve increased performance and parallelism. *See*, *e.g.*, Ex. 2016 ("BF16 delivers a rare combination: reducing hardware and energy while simplifying software by making loss scaling unnecessary;" *see also* Ex. 2011 (touting the increased efficiency and performance achieved by using low-precision multipliers).

IX. GOOGLE'S TPUV2 AND TPUV3 ARE COEXTENSIVE WITH THE CHALLENGED CLAIMS

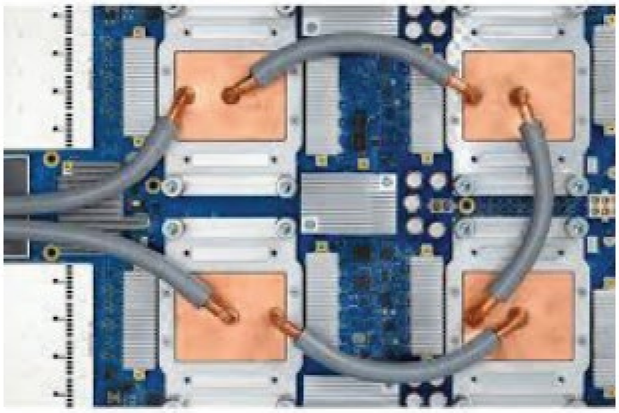
Google's TPUv2 and TPUv3 are coextensive with claims 1-8 of the '156 Patent. Set forth below is an analysis mapping the claims to the TPUv2 and TPUv3 devices; a more fulsome analysis is also provided in Singular's district court infringement contentions. Exs. 2021; 2022, 1-12. Accordingly, a nexus is presumed between secondary considerations relating to the TPUv2 and TPUv3 and the '156 Patent. *WBIP*, 829 F.3d at 1329 ("[T]here is a presumption of nexus for objective considerations when the patentee shows that the asserted objective evidence is tied to a specific product and that product 'is the invention disclosed and claimed in the patent.'").

A. Claim 1

1. "A device comprising:"

Each of the TPUv2 boards and TPUv3 boards are devices. Khatri, ¶¶ 129-30.

TPUv2 Board - Ex. 2016.

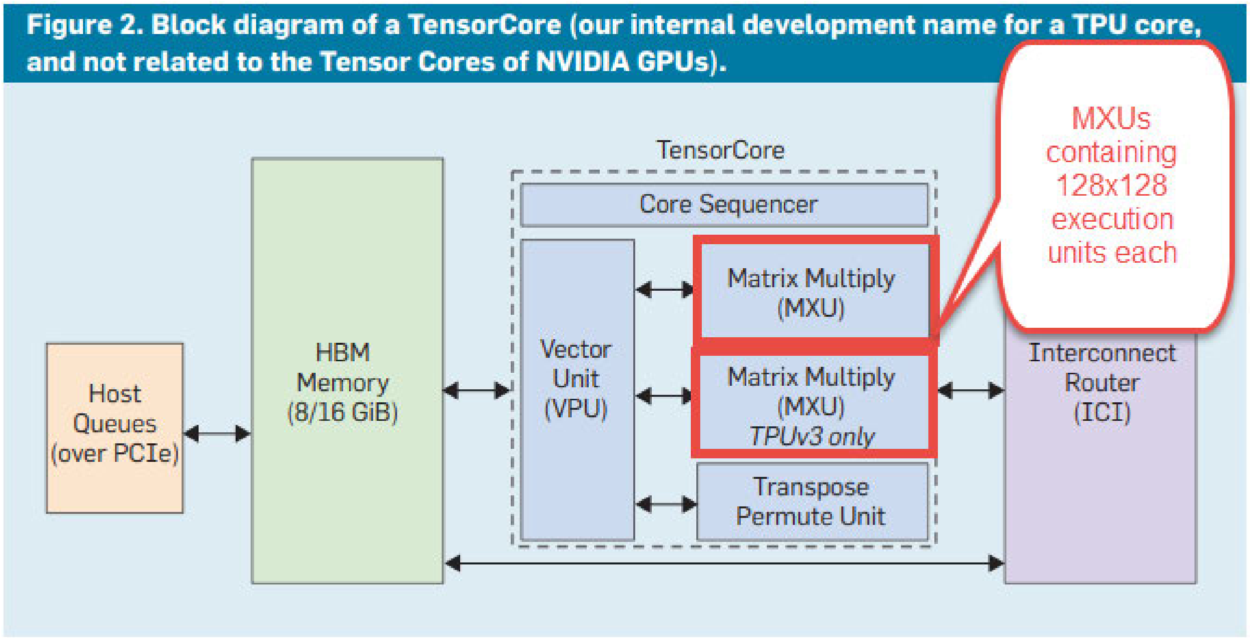


TPUv3 Board Ex. 2016.

2. “at least one first low precision high dynamic range (LPHDR) execution unit adapted to execute a first operation on a first input signal representing a first numerical value to produce a first output signal representing a second numerical value”

Each of the TPUv2 and TPUv3 boards include at least one first low precision high dynamic range (LPHDR) execution unit adapted to execute a first operation (multiplication) on a first input signal representing a first numerical value to produce a first output signal representing a second numerical value (the product of the first numerical value multiplied by another value). Khatri, ¶ 131. Both versions include at least one Matrix Multiply Unit (MXU) that contains 128x128 = 16,384 execution units that each execute the operation of multiplication. Exs. 2016; 2011 (“the main computational capacity in each core provided by a large matrix multiply unit that can yield the results of multiplying a pair of 128x128 matrices each cycle”).

Figure 2. Block diagram of a TensorCore (our internal development name for a TPU core, and not related to the Tensor Cores of NVIDIA GPUs).



Ex. 2016 (annotations added).

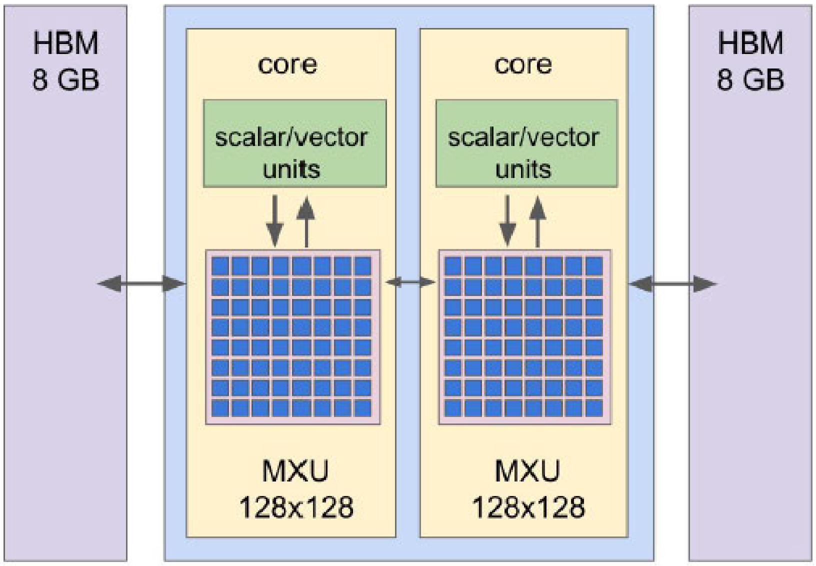


Figure 5: A block diagram of Google's Tensor Processing Unit v2 (TPUv2)

Ex. 2011.

Each of the bfloat16 execution units in the MXU require "less circuitry" than either FP32 or FP16 multipliers. Ex. 2011, 9; Khatri, ¶ 132. Accordingly, each of the multipliers within the MXU is *only* able to perform low-precision, bfloat16 operations and lacks the circuitry to perform, for example, perform FP16 or FP32 operations. Khatri, ¶ 132.

3. “wherein the dynamic range of the possible valid inputs to the first operation is at least as wide as from 1/65,000 through 65,000 and”

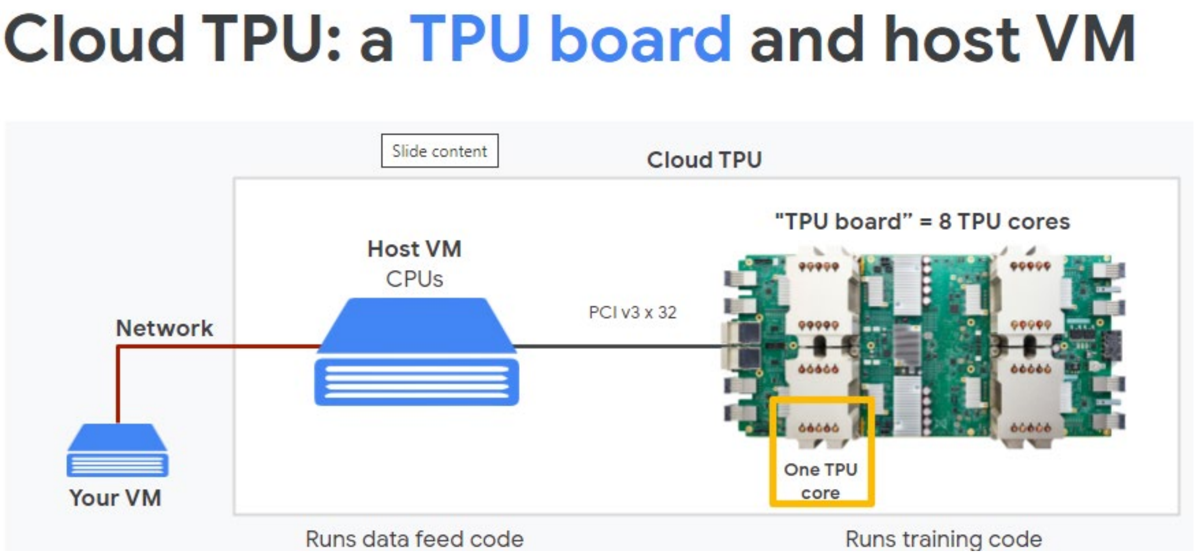
The dynamic range of the valid inputs to the multiplication operation in the TPUv2 and TPUv3 is governed by the number of exponent bits (8) in the floating point 32 format. Khatri, ¶¶ 133-34; Pet., 20-21; Ex. 1003, ¶ 231. These inputs are supplied by the “scalar/vector units,” also known as Vector Processing Unit (VPUs). *E.g.*, Exs. 2016; 2049, 3-4. As Google admits, 8 bits of exponent allow for a dynamic range of from roughly 2^{-126} (smaller than 1/65,000) through 2^{127} (larger than 65,000). Khatri, ¶134; Pet., 20-21; Ex. 1003, ¶ 232.

4. “for at least X=5% of the possible valid inputs to the first operation, the statistical mean, over repeated execution of the first operation on each specific input from the at least X% of the possible valid inputs to the first operation, of the numerical values represented by the first output signal of the LPHDR unit executing the first operation on that input differs by at least Y=0.05% from the result of an exact mathematical calculation of the first operation on the numerical values of that same input;”

Google’s TPUv2 and TPUv3 boards meet this element by performing multiplication on 32-bit numbers at bfloat16 precision, which uses 7 bits for the mantissa. Exs. 2041, 7; 2011, 8; 2049; Khatri, ¶ 135. Google states that utilizing 7 bits of mantissa in multiplication operations results in a minimum of 12% of valid floating point 32 inputs producing at least 0.39% relative error compared to the exact mathematical calculation of a full-precision multiplication on those same inputs. Pet., 73; Khatri, ¶¶ 135-36. Accordingly, the TPUv2 and TPUv3 meet this element.

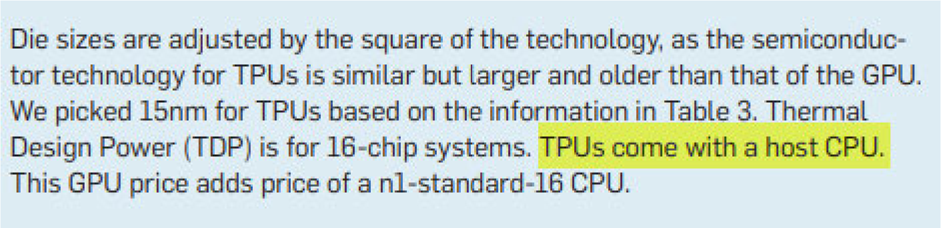
5. "at least one first computing device adapted to control the operation of the at least one first LPHDR execution unit."

Google's TPUv2 and TPUv3 devices meet this limitation as well. Each TPU device is controlled by a Core Sequencer (which, in turn, is controlled by a Host Virtual Machine that is a program running on a CPU). The Core Sequencer is a computing device, and by issuing VLIW instructions to the MXU, it controls the operation of the LPHDR execution units within the MXUs. Ex. 2016, 4-5. The CPU is a computing device, and by running the Host VM and issuing commands to the Core Sequencer, it controls the operation of the LPHDR execution units within the MXUs. Ex. 2016, 4; Khatri, ¶¶ 137-38.



Cloud TPU: a TPU board and host VM

Ex. 2045, 2.



Die sizes are adjusted by the square of the technology, as the semiconductor technology for TPUs is similar but larger and older than that of the GPU. We picked 15nm for TPUs based on the information in Table 3. Thermal Design Power (TDP) is for 16-chip systems. TPUs come with a host CPU. This GPU price adds price of a n1-standard-16 CPU.

Ex. 2011.

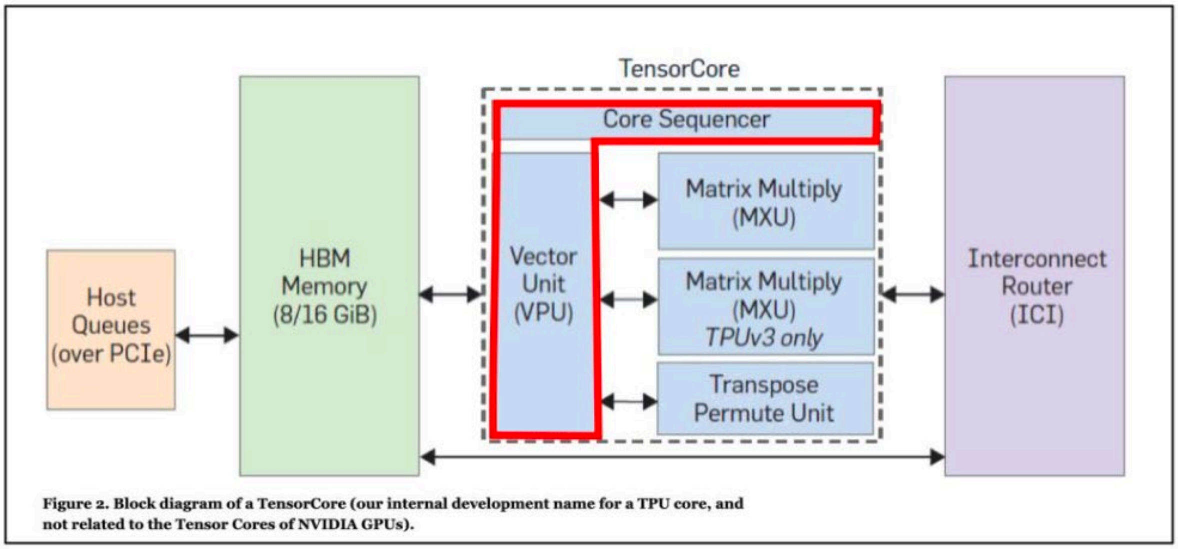


Figure 2. Block diagram of a TensorCore (our internal development name for a TPU core, and not related to the Tensor Cores of NVIDIA GPUs).

Ex. 2016, (annotations added)

B. Claim 2

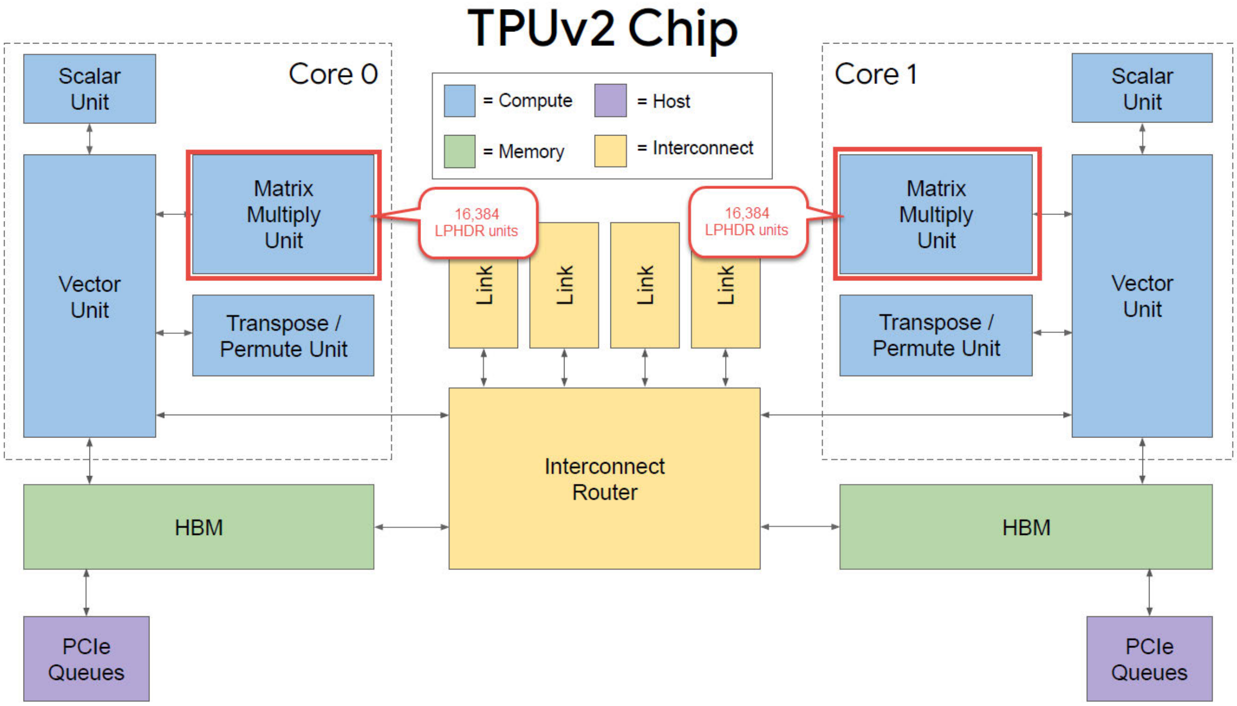
1. "The device of claim 1, wherein the at least one first computing device comprises at least one of a central processing unit (CPU), a graphics processing unit (GPU), a field programmable gate array (FPGA), a microcode-based processor, a hardware sequencer, and a state machine."

As discussed above, the Host VMs run on a CPU. The Core Sequencer is also a hardware sequencer and a state machine. Ex. 2016, 4; Khatri, ¶ 140. The Core Sequencer and the CPU, independently and acting in concert, satisfy the additional requirement imposed by claim 2. Khatri; ¶¶ 139-41.

C. Claim 3

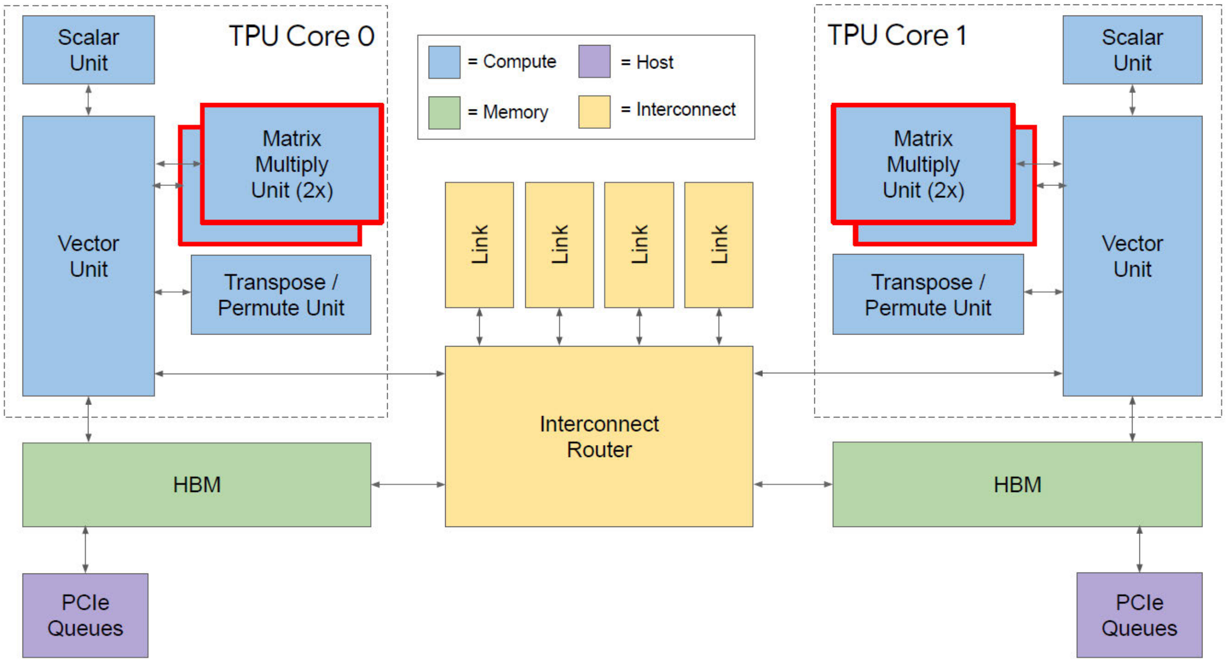
1. "The device of claim 2, wherein the number of LPHDR execution units in the device exceeds by at least one hundred the non-negative integer number of execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide."

Google's TPUv2 and TPUv3 boards meet this limitation. As discussed above, each MXU contains 128x128 execution units: a total of 16,384 LPHDR execution units in total per MXU. Khatri, ¶ 142. The TPUv2 board includes 1 MXU per core, with 2 cores in a chip, and 4 chips on a board. Ex. 2016. Thus, the TPUv2 board contains 131,072 LPHDR execution units.



TPUv2 Board – Ex. 2046 (annotations added)

The TPUv3 board contains 2 MXUs per core, 2 cores per chip, and 4 chips per board. Ex. 2016. Thus, a single TPUv3 board contains 262,144 LPHDR execution units. Khatri, ¶ 143.



TPUv3 Board – EX. 2046 (annotations added).

The MXU execution units perform multiplication only at bfloat16 precision and are thus not adapted to execute multiplication on floating point numbers that are at least 32 bits wide. Khatri, ¶ 144.

Each core (whether TPUv2 or TPUv3) contains a VPU. *Id.* Each VPU contains an array of 128x8x2 32-bit arithmetic logic units (ALUs), for a total of 2048 ALUs per core. Exs. 2016; 2046. 32-bit ALUs are adapted to perform the operation of multiplication on 32-bit numbers. *Id.*, ¶ 145. With two cores per chip and 4 chips per board, each TPUv2 or TPUv3 board contains 16,384 execution units that are adapted to perform multiplication on 32-bit numbers. *Id.*, ¶ 146.

Each TPUv2 board thus contains 131,072 LPHDR execution units and 16,384 VPUs. Khatri, ¶¶ 146-47. Each TPUv3 board contains 262,144 LPHDR execution units and 16,384 VPUs. *Id.* Accordingly, in each board "the number of LPHDR execution units in the device exceeds by at least one hundred the non-negative integer number of execution units in the device adapted to execute at least the operation of multiplication on floating point numbers that are at least 32 bits wide" and each board therefore meets claim 3. *Id.*

D. Claim 4

1. The device of claim 3, wherein X=10%.

See Section VIII.A.4 (mantissa of 7 bits gives at least 12% X). Khatri ¶ 148.

E. Claim 5

1. The device of claim 3, wherein Y=0.2%

See Section VIII.A.4 (mantissa of 7 bits gives at least 0.39% Y). Khatri ¶ 149.

F. Claim 6

1. The device of claim 3, wherein X=10% and Y=0.2%

See Section VIII.D and VIII.E (mantissa of 7 bits gives at least 12% X and 0.39% Y). Khatri ¶ 150.

G. Claim 7

1. The device of claim 3, wherein the dynamic range of the possible valid inputs to the first operation is at least as wide as from 1/1,000,000 through 1,000,000

As discussed above, the bfloat16 format used in the TPUv2 and TPUv3 boards contains 8 bits of exponent, which provides a dynamic range of from roughly 2^{-126} (smaller than 1/1,000,000) through 2^{127} (larger than 1,000,000). Khatri, ¶ 151.

H. Claim 8

1. The device of claim 3, wherein the first operation is multiplication.

The MXUs within the TPUv2 and TPUv3 boards perform multiplication. Khatri, ¶ 152.

X. CONCLUSION

For the foregoing reasons, Patent Owner respectfully requests that the Board issue a Final Written Decision confirming the patentability of all the Challenged Claims.

Respectfully submitted,

Dated: August 9, 2021 By: / *Peter Lambrianakos* /
Peter Lambrianakos (Reg. No. 58,279)
FABRICANT LLP
411 Theodore Fremd Avenue,
Suite 206 South
Rye, New York 10580
Tel. (212) 257-5797
Fax. (212) 257-5796

Email: plambriananankos@fabricantllp.com

CERTIFICATE OF WORD COUNT

The undersigned hereby certifies that the portions of the above-captioned SINGULAR COMPUTING LLC'S RESPONSE TO PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 9,218,156 specified in 37 C.F.R. § 42.24 has 13,492 words in compliance with the 14,000 word limit set forth in 37 C.F.R. § 42.24. This word count was prepared using Microsoft Word 2020.

Respectfully Submitted,

Dated: August 9, 2021

By: */ Peter Lambrianakos /*
Peter Lambrianakos (Reg. No. 58,279)
Lead Counsel for Patent Owner
FABRICANT LLP
411 Theodore Fremd Avenue,
Suite 206 South
Rye, New York 10580
Telephone: (212) 257-5797
Facsimile: (212) 257-5796
Email: plambrianakos@fabricantllp.com

CERTIFICATE OF SERVICE

A copy of SINGULAR COMPUTING LLC'S RESPONSE TO PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 9,218,156 and EXHIBITS 2026-2053 have been served on Petitioner's counsel of record as follows:

Elisabeth H. Hunt
Email: EHunt-PTAB@WolfGreenfield.com
Richard F. Giunta
Email: RGiunta-PTAB@WolfGreenfield.com
Anant K. Saraswat
Email: ASaraswat-PTAB@WolfGreenfield.com
Nathan R. Speed
Email: NSpeed-PTAB@WolfGreenfield.com
Gregory F. Corbett
Email: Gregory.Corbett@WolfGreenfield.com
WOLF, GREENFIELD & SACKS, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210

Attorneys for Google LLC

August 9, 2021

By: / *Peter Lambrianakos* /
Peter Lambrianakos (Reg. No. 58,279)
FABRICANT LLP
411 Theodore Fremd Avenue,
Suite 206 South
Rye, New York 10580
Telephone: (212) 257-5797
Facsimile: (212) 257-5796
Email: plambriananko s@fabricantllp.com